

Quantum Perceptron

Michael Siomau*

Physics Department, Jazan University, P.O. Box 114, 45142 Jazan, Kingdom of Saudi Arabia

(Dated: December 18, 2012)

The idea of information encoding on quantum bearers and its quantum-mechanical processing has revolutionized our world and brought mankind on the verge of enigmatic era of quantum technologies. Inspired by this idea, in present paper we search for advantages of quantum information processing in the field of machine learning. We show that the simplest learning machine – perceptron – can dramatically increase its learning capabilities, if operates according to the laws of quantum mechanics. Exploiting only basic properties of the Hilbert space, superposition principle of quantum mechanics and quantum measurements to introduce a quantum perceptron, we demonstrate, for instance, that it is able to learn an arbitrary (Boolean) logical function, while this learning task can not be performed by its classical counterpart. The quantum perceptron learning rule, moreover, does not require any optimization procedure, which is necessary for classical learning models.

PACS numbers: 03.67.-a, 87.19.1l, 87.19.lv

During last few decades, we have been witnessing unification of quantum physics and classical information science that resulted in constitution of new disciplines – quantum information and quantum computation [1]. While processing of information that is encoded in systems exhibiting quantum properties suggests, for example, unconditionally secure quantum communication [2] and superdense coding [3], computers that operate according to the laws of quantum mechanics offer efficient solving of problems that are intractable on conventional computers [4]. Having paramount practical importance, these announced technological benefits have indicated the main directions of the research in the field of quantum information and quantum computation, somehow leaving aside other potential applications of quantum physics in information science. So far, for instance, very little attention has been paid on possible advantages of quantum information processing in such areas of modern information science as machine learning [5] and artificial intelligence [6]. Although machine learning governed by quantum mechanics have been demonstrated to have certain advantages over classical learning [7, 8], these advantages are strongly coupled with more sophisticated optimization procedure than in the classical case. This paper, in contrast, presents a new approach for machine learning which does not require any optimization at all.

Our focus is on perceptron, which is the simplest learning machine. Perceptron is a model of neuron that was originally introduced by Rosenblatt [9] to perform visual perception tasks, which, in mathematical terms, result in solution of the linear classification problem. There are two essential stages of a perceptron functioning: supervised learning session and new data classification. During the first stage, the perceptron is given a labeled set of examples. Its task is of inferring weights of a linear function according to some error-correcting rule. Subse-

quently, this function is utilized for classification of new previously unseen data. In spite of its very simple learning rule and internal structure, perceptron's capabilities are seriously limited [10]. Perceptron can not provide the classification, if there is an overlap in the data or if the data can not be linearly separated. It is also incapable of learning complex logical functions, such as XOR function. Moreover, by its design, perceptron can distinguish only two classes and, therefore, can not resolve the situation when the input belongs to none of the two classes.

In this paper we show that all the mentioned problems can be, in principle, overcome by a quantum perceptron. There are also two operational stages for the quantum perceptron. During the learning stage all the data are formally represented through quantum states of physical systems. This representation allows expanding the data space to a physical Hilbert space. It is important to note, that there is no need to involve real physical systems during this stage. Thus, the learning is essentially a classical procedure. The subject of the learning is a set of positive operator valued measurements (POVM) [1]. The set is constructed by making superpositions of the training data in a way that each operator detects one particular class. This procedure is linear, does not require solving equations or optimizing parameters. When the learning is over, real quantum systems come into play: new data is encoded into the states of the quantum systems, which are measured with the POVM. Based on the results of the measurements, the required classification is achieved.

In the following, we shall briefly overview classical perceptron and discuss the origin of the restrictions on its learning capabilities. After this, we shall introduce a quantum perceptron and show how it can overcome the restrictions by example of XOR function learning. Finally, we shall discuss potential capabilities of the concept of quantum perceptron.

Operational structure of the classical perceptron is simple. Given an input vector \mathbf{x} (which is usually called a feature vector) consisting of n features, perceptron computes a weighted sum of its components $f(\mathbf{x}) = \sum_i a_i x_i$,

*Electronic address: m.siomau@gmail.com

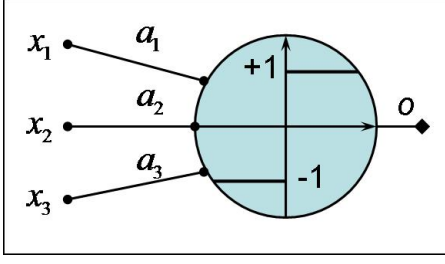


FIG. 1: A schematic representation of a classical perceptron with three input features.

where weights a_i have been previously learned. The output from a perceptron is given by $o = \text{sign}(f(\mathbf{x}))$, where sign is the Heaviside function

$$\text{sign}(y) = \begin{cases} +1 & y > 0 \\ -1 & y \leq 0 \end{cases}. \quad (1)$$

Depending on the binary output signal $o = \{+1, -1\}$, the input feature vector \mathbf{x} is classified between two feature classes, one of which is associated with output $o = +1$ and the other with output $o = -1$. A standard graphical representation of a perceptron is given in Fig. 1. As we have mentioned above, the perceptron needs to be trained before its autonomous operation. During the training, a set of P training data pairs $\{\mathbf{x}_i, d_i, i = 1, \dots, P\}$ is given, where \mathbf{x}_i are the n -dimensional feature vectors and d_i are desired binary outputs. Typically, at the beginning of the learning procedure the initial weights a_i of the linear function are generated randomly. When a data pair is chosen from the training set, the output $o_i = \text{sign}(f(\mathbf{x}_i))$ is computed from the input feature vector \mathbf{x}_i and is compared to the desired output d_i . If the actual and the desired outputs match $o_i = d_i$, the weights a_i are left without change and the next pair from the data set is taken for the analysis. If $o_i \neq d_i$, the weights a_i of the linear function are to be changed according to the error-correcting rule $\mathbf{a}' = \mathbf{a} + \epsilon \mathbf{a} = \mathbf{a} + (d_i - o_i)\mathbf{x}_i$. The error-correcting rule is applied until the condition $o_i = d_i$ is met.

The training procedure has a clear geometric interpretation. The weights a_i of the linear function define a $n - 1$ -dimensional hyperplane in the n -dimensional feature space. The training procedure results in a hyperplane that divides the feature space on two subspaces, so that each feature class occupies one of the subspaces. Due to this interpretation, the origin of the restrictions on learning capabilities of the classical perceptron becomes visible: a hyperplane that separates the two classes may not exist. An example of two classes that can not be linearly separated is XOR logical function of two variables

$$\begin{array}{cccc} x_1 & 0 & 0 & 1 & 1 \\ x_2 & 0 & 1 & 0 & 1 \\ f & 0 & 1 & 1 & 0 \\ o & -1 & +1 & +1 & -1 \end{array}, \quad (2)$$

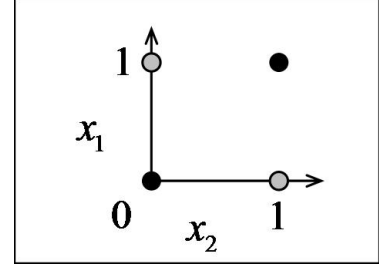


FIG. 2: The feature space of XOR function is two-dimensional and discrete (each feature takes only values 0 and 1). There is no a line (a one-dimensional hyperplane) that separates 0s and 1s. Classical perceptron is incapable of classifying the input feature vectors and, therefore, can not learn XOR function.

A representation of this function in the two-dimensional feature space is shown in Fig. 2.

Understanding the principles of classical perceptron functioning, we are ready to move forward and introduce quantum perceptron. In order to simplify our discussion, let us first consider a particular classification task – XOR function learning.

As its classical counterpart, quantum perceptron is to be trained to perform the classification task. Suppose, we are given a set of four training data pairs $\{\mathbf{x}_i, d_i, i = 1, \dots, 4\}$, where the feature vector consists of two features $\mathbf{x} = \{x_1, x_2\}$, and the desired output $d = \{+1, -1\}$ is a binary function. Let us represent the input features through the states of a two-dimensional quantum system – qubit, so that each feature is given by one of the basis states $|x_i\rangle = \{|0\rangle, |1\rangle\}$ for $i = 1, 2$, where $\{|0\rangle, |1\rangle\}$ denotes an orthonormal computational basis [1]. The quantum representation allows extending the two-dimensional feature space to four-dimensional Hilbert space of the two-qubit system. In the above representation, the feature vector \mathbf{x} is given by one of the four two-qubit states $|x_1, x_2\rangle$.

During the learning, for a given feature vector $|x_1, x_2\rangle$ and desired output d , let us find a vector $|\psi\rangle$ from the condition $\langle\psi|x_1, x_2\rangle = |d|$ and construct an operator $P_d = |\psi\rangle\langle\psi|$, where the modulus $|d|$ is taken in order to avoid construction of unphysical (negative) operators. Repeating this procedure for all data from the training data set, let us summate and normalize all operators that belong to $d = -1$ and $d = +1$. In result of the four data pairs learning, we have two operators

$$\begin{aligned} P_{-1} &= |0, 0\rangle\langle 0, 0| + |1, 1\rangle\langle 1, 1|, \\ P_{+1} &= |0, 1\rangle\langle 0, 1| + |1, 0\rangle\langle 1, 0|. \end{aligned} \quad (3)$$

It is easy to check that operators P_{-1} and P_{+1} are orthogonal $P_{-1}P_{+1} = 0$ and form a complete set $P_{-1} + P_{+1} = I$, where I is the identity operator. During its autonomous operation, quantum perceptron may be given a two-qubit system prepared in one of the four states $|x_1, x_2\rangle = \{|0, 0\rangle, |0, 1\rangle, |1, 0\rangle, |1, 1\rangle\}$. With the help of

the operator P_{-1} states $|0, 0\rangle$ and $|1, 1\rangle$ are measured and assigned class $d = -1$, while the states $|0, 1\rangle$ and $|1, 0\rangle$ are detected by the operator P_{+1} and classified to $d = +1$. The fact that the operators P_{-1} and P_{+1} are orthogonal ensures zero probability of misclassification, while completeness of the set of operators guarantees classification of any input. Conclusively, the quantum perceptron has learned XOR function.

The successful XOR function learning by quantum perceptron is the consequence of the representation of the classical feature vector \mathbf{x} through the two-qubit states. In the classical representation, the feature vectors can not be linearly separated on a plane, see Fig. 2. In the quantum representation, four mutually orthogonal states $|x_1, x_2\rangle$ in the four-dimensional Hilbert space can be separated on two classes in an arbitrary fashion. This implies that an arbitrary logical function of two variables can be learned by quantum perceptron. For example, learning of logical AND function leads to the construction of operators $P_{-1} = |0, 0\rangle\langle 0, 0| + |0, 1\rangle\langle 0, 1| + |1, 0\rangle\langle 1, 0|$ and $P_{+1} = |1, 1\rangle\langle 1, 1|$. Moreover, an arbitrary logical function of an arbitrary number of inputs also can be learned by quantum perceptron, because the number of inputs of such a function growth exponentially with the order of the function and exactly as fast as dimensionality of the Hilbert space that is needed to represent the logical function. It is very important to note that, in spite of the exponential growth of the Hilbert space, the number of qubits, that are required for the learning, growth linearly and there are always just two (or, as we will see later, three) operators to perform during autonomous work of the perceptron.

Form the above example we have seen how the quantum representation helps to learn the logical XOR function, but the role of quantum measurements was not evident. Let us slightly modify the problem of XOR learning to explain the reason of introducing quantum measurements in the quantum perceptron functioning. In real-life learning tasks the training data may be corrupted by noise [5]. In some cases, noise may lead to overlapping of the training data, which result in misclassification of feature vectors during the training stage and during further autonomous functioning. For example, if, during the XOR learning, there is a finite small probability δ that feature x_1 takes a wrong binary value, but the other feature and the desired output are not affected by noise, after a big number of trainings (which are usually required in case of learning from noisy data), the operators P_{-1} and P_{+1} are given by

$$\begin{aligned} P'_{-1} &= (1 - \delta) (|0, 0\rangle\langle 0, 0| + |1, 1\rangle\langle 1, 1|) \\ &\quad + \delta (|0, 1\rangle\langle 0, 1| + |1, 0\rangle\langle 1, 0|), \\ P'_{+1} &= (1 - \delta) (|0, 1\rangle\langle 0, 1| + |1, 0\rangle\langle 1, 0|) \\ &\quad + \delta (|0, 0\rangle\langle 0, 0| + |1, 1\rangle\langle 1, 1|). \end{aligned} \quad (4)$$

These operators P'_{-1} and P'_{+1} form a complete set $P'_{-1} + P'_{+1} = I$, but they are not orthogonal any more

$P'_{-1}P'_{+1} \neq 0$. This means that during autonomous operation of the quantum perceptron, the input feature vectors are misclassified with probability δ . Nevertheless, on average, most of the feature vectors are classified correctly. This means that quantum perceptron simulates XOR function with a degree of accuracy given by $1 - \delta$. At this point one may think of an error-correcting procedure to transform operators P'_{-1} and P'_{+1} to operators P_{-1} and P_{+1} , which may be an orthogonalization of two diagonal operators. But, we prefer to look at the result of the learning from the noisy training data from a slightly different perspective. When quantum perceptron is trained on the noisy data, it can exactly (in probabilistic sense) reproduce fluctuations that have been observed during the training. This ability may be useful in some cases. At least, classical perceptron can not do anything like this.

From the above discussion we can conclude that, having internally probabilistic nature, quantum measurements allow probabilistic classifying of the feature vectors, when there is some nonzero probability of misclassification. Rephrasing the last statement, due to quantum measurements, quantum perceptron is capable of performing classification when there is an overlap in the data.

Now, we have a sufficient background to analyze the full power of the concept of quantum perceptron. Let us have a closer look on the learning stage. Given a set of P training data pairs $\{\mathbf{x}_i, d_i, i = 1, \dots, P\}$, we need to represent the input features through the states of a quantum system. This is indeed the most difficult part. In the case of Boolean functions the quantum representation through qubits is intuitively understandable. In general, classification tasks may be very different in origin; therefore we do not have a general receipt to construct a quantum representation for a given (real) n -dimensional feature space R^n . The main requirement for the construction is that the quantum representation must have all the topological properties of the original feature manifold.

But, suppose we have constructed a quantum representation and found operators $P_d = |\psi\rangle\langle\psi|$ according to the rule $\langle\psi|x_1, x_2\rangle = |d|$. Making the sum of the corresponding operators we eventually get two operators P_{-1} and P_{+1} . Each of these operators take into account all training data that belongs to the corresponding output. Therefore, by their construction and due to the superposition principle of quantum mechanics, operator P_{-1} can detect any previously seen feature vector (that corresponds to the output $o = -1$) and an arbitrary linear (convex) combination of such feature vectors. The same applies to the operator P_{+1} . There are only four possibilities of how learning stage may end:

- Operators P_{-1} and P_{+1} are orthogonal $P_{-1}P_{+1} = 0$ and form a complete set $P_{-1} + P_{+1} = I$. We have met this case when analyzing XOR learning without noise: there is no overlap in the data and each input feature vector can be classified between the two classes with no mistake.

- Operators P_{-1} and P_{+1} are orthogonal $P_{-1}P_{+1} = 0$, but do not form a complete set $P_{-1} + P_{+1} \neq I$. This is an extremely interesting case. We can define the third operator $P_0 = I - P_{-1} - P_{+1}$, which is orthogonal to P_{-1} and P_{+1} , because $P_{-1}P_{+1} = 0$. During its autonomous functioning, quantum perceptron generates three outputs $d = \{+1, 0, -1\}$, namely that the feature vector belongs to the one of the previously seen classes $d = \{+1, -1\}$ or it is essentially different from the learned classes $d = 0$ – it belongs to a new, previously unseen, class [11]. The classification on previously unseen classes is an extremely hard learning problem, which can not be done even by the most of the classical perceptron networks [5, 10]. Quantum perceptron is capable of performing this task with no mistakes due to the orthogonality of the operators P_{-1} , P_{+1} and P_0 .
- Operators P_{-1} and P_{+1} are not orthogonal $P_{-1}P_{+1} \neq 0$, but form a complete set $P_{-1} + P_{+1} = I$. This is the case of the noisy XOR learning: all the data can be classified on the two classes with some nonzero probability of mistake.
- The most general case is when operators P_{-1} and P_{+1} are not orthogonal $P_{-1}P_{+1} \neq 0$ and do not form a complete set $P_{-1} + P_{+1} \neq I$. We can again define the third operator $P_0 = I - P_{-1} - P_{+1}$, which this time is not orthogonal to P_{-1} and P_{+1} . In this situation, quantum perceptron classifies all the input feature vectors on three classes, one of which is a new class, with some nonzero probability of mistake.

When the training stage is over, the only possible restriction on practical implementations of quantum perceptron is our ability to prepare and measure quantum systems. In the case of logical functions learning we need to deal with qubits, which may be electrons, nuclear or molecular spins, quantum dots or photons. Being at the heart of quantum information science, the art of prepara-

tion and detection of such systems has reached unprecedented heights. Current technologies, for example, allow us handling 10^5 photonic qubits [12]. Therefore, we do not see practical limitations on implementation of quantum perceptron, at least, for the task of logical functions learning.

In conclusion, bridging between quantum information science and machine learning theory, we showed how capabilities of a learning machine can be dramatically increased, if it operates according to the laws of quantum mechanics. We introduced quantum perceptron and argued that it can potentially perform tasks, which are un-doable for its classical counterpart: learning arbitrary logical functions, classification of data with an overlap and classification on previously unseen classes. We supported our point showing explicitly how quantum perceptron can learn logical XOR function.

It is very important to note that learning of complex logical functions and classify data with an overlap can be performed in the framework of classical learning models, for example, by support vector machines [5]. However, implementation of any classical model demands optimization, which complicates rapidly with growth of the feature space, the so-called curse of dimensionality. Quantum perceptron is immune to the curse, since its learning rule does not require any optimization. Moreover, none of the classical learning models can build the third class from the data that do not belong to the two classes observed during the learning stage, while quantum perceptron perform this task with no difficulties.

As a final remark we would like to note that a network of classical perceptrons (with a hidden layer) is shown to be a universal approximator [13]. This implies that an arbitrary real-valued function can be simulated by the network and, therefore, the network is as powerful (in computational capabilities) as Turing machine. It may be that a quantum perceptron network is competitive with a quantum computer, if not exceeds it in computational power.

-
- [1] M.A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information*, (Cambridge University Press, Cambridge, 2000).
 - [2] N. Gisin *et al.*, Rev. Mod. Phys. **74**, 145 (2002).
 - [3] V. Vedral, Rev. Mod. Phys. **74**, 197 (2002).
 - [4] A. M. Childs and W. van Dam, Rev. Mod. Phys. **82**, 1 (2010).
 - [5] V. Kecman, *Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models*, (MIT Press, Cambridge, 2001).
 - [6] S.J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, (Prentice Hall, New Jersey, 2009).
 - [7] R. Zhou, L. Qin and N. Jiang, LNCS **4131**, 651 (2006).
 - [8] D. Manzano, M. Pawłowski and C. Brukner, New. J. Phys. **11**, 113018 (2009).
 - [9] F. Rosenblatt, "The Perceptron - a perceiving and recognizing automaton" (Rep. 85-460-1, Cornell Aeronautical Laboratory, New York, 1957).
 - [10] M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry*, (MIT Press, Cambridge, 1969).
 - [11] A situation when $P_{-1}P_{+1} = 0$ and $P_{-1} + P_{+1} \neq I$ may essentially appear during noiseless XOR learning. Suppose, we are given just three feature vectors $|x_1, x_2\rangle = \{|0, 0\rangle, |0, 1\rangle, |1, 0\rangle\}$ during the training stage. Based on these incomplete training data the operators P_{-1} and P_{+1} are constructed as $P_{-1} = |0, 0\rangle\langle 0, 0|$ and $P_{+1} = |0, 1\rangle\langle 0, 1| + |1, 0\rangle\langle 1, 0|$. The third operator needs to be defined as $P_0 = |1, 1\rangle\langle 1, 1|$ to complete the POVM set.
 - [12] T.Sh. Iskhakov *et al.*, Phys. Rev. Lett. **109**, 150502 (2012).
 - [13] K. Hornik, Neural Networks, **4**, 251 (1991).